

APPLICATION FOR LETTERS PATENT OF THE UNITED STATES

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

"EXPRESS MAIL" Mailing Label Number EM124252854 US

Date of Deposit MAR. 31, 2000

I hereby certify that this paper or fee is being deposited with the United States Postal Service "EXPRESS MAIL POST OFFICE TO ADDRESSEE" Service under 37 CFR 1.10 on the date indicated above and is addressed in the Assistant Commissioner for Patents, Washington, d.c. 20231

DORIS WILLIAMS
(Typed or Printed Name of Person Mailing Paper or Fee)

Doris Williams
(Signature of Person Mailing Paper or Fee)

SPECIFICATION

To all whom it may concern:

Be It Known, That we, THOMAS G. BREWER and NELSON H. YOUNT, of Lexington, SC and Lexington, SC, respectively, have invented certain new and useful improvements in CHANNEL DIRECTOR FOR CROSS-CHANNEL CUSTOMER INTERACTIONS, of which we declare the following to be a full, clear and exact description:

CHANNEL DIRECTOR FOR CROSS-CHANNEL CUSTOMER INTERACTIONS

Thomas G. Brewer

Nelson H. Yount

5

FIELD OF THE INVENTION

The present invention relates to methods and an apparatus used to capture and evaluate concurrent electronic interactions occurring between a customer and a service provider over multiple communication channels, thereby enhancing the interactions and the channels.

10

BACKGROUND OF THE INVENTION

Businesses have traditionally attempted to automate and customize the way in which they interact with a customer. Customers contact a business through a variety of media. Often these media are referred to as “channels,” some of these channels include customer interactions made via: email, cable, satellite, fax, kiosk, Internet, or directly in person (hereinafter “walk-up”). Software is often deployed to transform customer contacts into interactions which are electronic representations of the communications occurring with a customer during a contact with a business. This may be accomplished by transcribing the communications by a business representative either during or after the contact, or may be readily apparent to a business software process when the customer is communicating through a business provided interface or device, such as an automated telephone menu or an electronic form. Converting these customer contacts into an

15

20

25

electronic media permits businesses to evaluate and improve future customer contacts.

By improving the way in which a business interacts with a customer over a channel, the business improves its customer relationship thereby increasing the likelihood of retaining an existing customer, and potentially generating additional business from the customer. This process of improving and customizing customer relationships is often referred to as “customer intimacy.”

Businesses have also sought to reduce operating expense associated with customer communications. Believing that technology can assist in achieving customer intimacy as well as reducing operating expenses, businesses have developed a variety of technologies to personalize and automate customer interactions with the business. Some of these technologies include automated telephone support, automatically generating an email response, and world wide web (WWW) form processing.

Furthermore, businesses have recently sought to take advantage of the tremendous activity on the WWW by attempting to capture as much information about their existing or potential customers as possible and then warehousing this profile information in an electronic database. Intelligent software agents are being developed to mine the warehouse so as to create focused marketing and interactions with the customers. However, businesses are now recognizing that they have multiple databases with customer profile information and no cost effective means to have these disparate databases communicate with each other so as to present a comprehensive view of a single customer. Additionally, efforts seem to be disproportionately focused on interactions with customers which occur over a WWW channel. Along these same lines, business policies are difficult to

enforce consistently when customers interact with a business over a variety of channels because the technology does not inter-operate well enough to permit consistent treatment of cross channel interactions with the customer.

An additional consideration impacting a business's efforts to capture
5 customer profile information is the concern for customer privacy. As a
result of this concern, federal agencies have begun developing privacy
guidelines associated with electronic privacy. Additionally, some foreign
countries have enacted stringent laws with respect to customer privacy.
Businesses must develop applications which are flexible enough to respond
10 to this growing tide of public sentiment.

Still, some attempts have been made at channel integration (often
referred to as "media blending"). These attempts focus on publishing
customer interactions across several channels upon completion of an
interaction with no attempt made to capture multiple concurrent interactions
15 with the customer on cross channels. Thus, prior methods do not permit
near-term use of the customer interactions to enhance the current customer
interaction. Further, these prior methods fail to capture all customer
interactions occurring over cross channels. These failures result in
customer interactions which fall between the cracks. The result of this is
20 that businesses are not capable of presenting a comprehensive 360 degree
view of their customers.

Finally, the explosion of technology and the ease of electronic
communication have created an added pressure on businesses to provide to
its customers all possible channels within which the customer can contact
25 the business. Yet, the addition of new channels create stovepipes of
operations that complicate managing the customer relationship and add
overhead expense associated with the maintenance and support of the new

channels of communication. Additionally, business have found it exceedingly difficult to connect customers to the appropriate business service provider as the subject matter and the channel associated with a customer interaction changes.

5 Accordingly, it is apparent that methods and an apparatus that permit near-term responses to a customer based upon cross channels of communication will allow for a closed-loop view of the customer and prevent interactions from falling between the cracks. This will engender greater customer intimacy by providing a comprehensive view of the
10 customer (360 degrees). Further, it is apparent that any such method or apparatus must be mindful of business policies, and customer privacy.

SUMMARY OF THE INVENTION

 Accordingly, it is an object of the present invention to overcome prior methods and/or apparatus to provide cross-channel customer
15 interactions which are captured and evaluated so as to modify at least one of the channels or interactions occurring with the customer. It is a related object to modify at least one of the channels or interactions while the customer remains in contact with the business during a session with the business, so as to provide near-term responses to the customer. It is a
20 further object of the present invention to permit customer interactions to be stored in a customer profile database. Customers can be identified by extracting information from the customer profile database and a business service provider identified. The service provider can then initiate an interaction with the customer. A final object of the present invention
25 permits the cross-channel interactions of a customer to be connected with

business service providers based upon of the subject matters and channels associated with the interactions.

To achieve the forgoing and other objects, a method for managing the cross-channel interactions of a customer are provided. The method
 5 comprises identifying a first interaction on a first channel and a second interaction on a second channel, identifying a customer associated with the interactions; recording a dialogue corresponding to the first and second interactions, and using the dialogue to modify at least one of the interactions.

10 In a preferred embodiment, shown in Fig. 1, an external customer 75 initiates multiple contacts with a business through a first channel₁ 320 and a second channel₂ 330. A Contact Manager 20 detects the contacts and determines if the contacts are existing interactions switching channels or new interactions. If the contacts are existing interactions switching
 15 channels then the interactions are updated to reflect the new channels over which the interactions are occurring. If the contacts are new interactions, the first channel₁ 320 is associated with a first interaction₁ 340, and the second channel₂ 330 is associated with a second interaction₂ 380. These two interactions are captured as a dialogue 60. A Connection Manager 140
 20 assigns a first business service provider₁ 1160 to the first interaction₁ 340 and a second business service provider₂ 1170 to the second interaction₂ 380. The dialogue 60 is then evaluated so as to modify interaction₁ 340 to establish interaction₁' 350 and interaction₂ 380 is modified to establish interaction₂' 390. Likewise, channel₁ 320 is modified to reflect channel₁'
 25 360 and channel₂ 330 is modified to reflect channel₂' 370.

In a further embodiment, the invention is directed to apparatus for managing cross-channel interactions of a customer. The apparatus

comprises a processor, a customer, a plurality of devices interfaced to the processors, means for identifying a first interaction on a first channel of one of the devices and a second interaction on a second channel of another of the devices, means for recording a dialogue corresponding to the first and
 5 second interactions, means for associating the customer with the first and second interactions, and means for using the dialogue to modify at least one of the interactions.

In the preferred embodiment, again referring to Fig. 1, an external customer 75 establishes a connection to a business through device₁ 400 and
 10 device₂ 410. Device₁ 400 will translate through a contact manager 20 to channel₁ 320 while device₂ 410 will translate to channel₂ 330. The remaining description of this embodiment is discussed above with the initial reference to Fig. 1.

In yet another object of this invention, a method is provided for
 15 managing the cross-channel interactions of a customer while the interactions are occurring. The method comprises: establishing a session including a first interaction on a first channel and a second interaction on a second channel, identifying a customer associated with the interactions; recording a dialogue associated with the first interaction and the second
 20 interaction during the session, and modifying at least one of the interactions using the dialogue before the session terminates.

Fig. 1 illustrates a session 420 for the preferred embodiment of this invention. The session 420 illustrates a period of time during which an external customer 75 and a business are interacting. During the session
 25 420, interaction₁ 340 and channel₁ 320 are modified as interaction₁' 350 and channel₁' 360 based on a dialogue 60.

A further object provides a method for customizing customer interactions which are initiated from a service provider. The method comprises providing a customer profile database, storing in the customer profile database a plurality of dialogues occurring with a customer,
5 initiating a service provider, and establishing an interaction initiated by the service provider with a customer extracted from the customer profile database wherein the interaction occurs over a channel.

In a preferred embodiment, Fig. 2, a service provider is initiated in step 230. Dialogues in step 290 are recorded in a customer profile database
10 in step 250. The initiated service provider in step 230 extracts a customer from the customer profile database in step 266. In step 270 an interaction on a channel is created, and in step 280 the interaction is updated with the identity of a particular customer with whom the service provider wishes to interact. In step 290 a connection is made to the customer. The interaction
15 then proceeds between the service provider and the customer with actions that are taken by the service provider validated in step 300. Finally, in step 310 the entire interaction is used to update the customer profile database.

A remaining object of this invention provides a method for connecting a customer interaction to a service provider based upon the
20 interaction and the subject matter of the interaction. The method comprises establishing a first interaction on a first channel and a second interaction on a second channel with a customer, identifying a first subject matter associated with the first interaction and a second subject matter associated with the second interaction, selecting a first service provider for the first
25 interaction based upon the first subject matter and the first channel, selecting a second service provider for the second interaction based upon the second subject matter and the second channel, connecting the first

service provider to the customer through the first interaction, and connecting the second service provider to the customer through the second interaction.

In a preferred embodiment, Fig. 3, a first contact is identified in step 1180 and second contact is identified in step 1190. The identify of a first channel and a second channel are determined in steps 1200 and 1210 respectively. A first interaction is created in step 1220 and a second interaction in step 1230. The interactions are tracked in a dialogue in step 1240. The customer associated with the interactions and the dialogue is located in step 1270. Once the customer is identified the interactions are updated to reflect this in step 1280. As the dialogue proceeds the identity of first subject matter associated with the first interaction will be identified in step 1250. Correspondingly, a second subject matter associated with the second interaction will be identified in step 1260. Based upon the subject matter of the interactions, first and second service providers are selected to communicate with the customer in steps 160 and 170 respectively.

Still other objects of the present invention will become apparent to those skilled in this art from the following description wherein there are shown and described preferred embodiments of this invention, simply for purposes of illustration. As will be realized, the invention may take on other aspects and arrangements than those described in detail below without departing from scope of the invention, as defined by the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

While the specification concludes with claims particularly pointing out and distinctly claiming the invention, it is believed that the invention

will be better understood from the following description taken in conjunction with the accompanying drawings in which:

Fig. 1 illustrates a preferred embodiment of the present invention;

Fig. 2 illustrates a flow diagram depicting the steps in an embodiment of the present invention showing methods wherein a service provider initiates an interaction with a customer;

Fig. 3 illustrates a flow diagram depicting the steps in an embodiment of the present invention showing methods wherein a service provider is connected to an interaction;

Fig. 4 illustrates a preferred architecture of the present invention;

Fig. 5 illustrates a hierarchical view and methods associated with a Channel Director Object in accordance with the present invention;

Fig. 6 illustrates a hierarchical view and methods associated with a Contact Manager Object in accordance with the present invention;

Fig. 7 illustrates a hierarchical view and methods associated with an Interaction Object in accordance with the present invention.

Fig. 8 illustrates a hierarchical view and methods associated with a Dialogue Object in accordance with the present invention;

Fig. 9 illustrates a hierarchical view and methods associated with a Content Analysis Object in accordance with the present invention;

Fig. 10 illustrates a hierarchical view and methods associated with a Customer Object in accordance with the present invention;

Fig. 11 illustrates the methods associated with a Channel Object in accordance with the present invention;

Fig. 12 illustrates a hierarchical view and methods associated with a Service Provider Object in accordance with the present invention;

Fig. 13 illustrates a hierarchical view and methods associated with a Business Policy Object in accordance with the present invention; and

Fig. 14 illustrates a hierarchical view and methods associated with a Connection Manager Object in accordance with the present invention;

5

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

The preferred operating system of this invention is Microsoft Corporation's Windows NT operating system, and the preferred programming language used to implement the present invention is C++.

10 The preferred programming method used to implement the present invention is Object Oriented Analysis and Programming (OOAP).

This invention is best understood by those skilled in the art through OOAP techniques as opposed to traditional structured programming techniques. The OOAP techniques will permit the objects to support data
15 abstraction/encapsulation, polymorphism, and inheritance. This technique permits real world entities to be encapsulated and abstracted into software representations called objects. Once the interfaces and methods of these objects are known, they may be assembled in a variety of structured ways to achieve results.

20 In a preferred embodiment of this invention the primary real world entities represented with software objects comprising Interaction Objects 30, Dialogue Objects 60, Customer Objects 70, and Service Provider Objects 100. Interaction Objects 30 are software representations of communications occurring during a business contact. They may be
25 automatically detected in the situation where the contacts are using business provided interfaces, or they may be transcribed into a business provided interface by a business representative as they are occurring. Dialogue

Objects 60 are software representations of multiple interactions occurring with a single customer. Customer Objects 70 are software representations of external customers, some of which may initially be anonymous. Service Provider Objects 100 are software representations of business representatives. These may be representations of human entities or software entities such as intelligent agents designed to do automated tasks such as generating email responses to customer requests.

Figs. 1 through 4 illustrate a preferred embodiment of the architecture and the object interfaces in accordance with the present invention. Fig. 1 illustrates a first interaction₁ 340 and a second interaction₂ 380 occurring over a first channel₁ 320 and a second channel₂ 380 recorded in a Dialogue Object 60. The Dialogue Object 60 is evaluated to modify the first interaction₁ 340 and the second interaction₂ 380 producing interaction₁ 250 and interaction₂, respectively. The Dialogue Object 60 is also evaluated to modify the first channel₁ 320 and the second channel₂ 330 producing channel₁ 360 and channel₂ 370, respectively. Further, an external customer 75, makes contact to one of the devices (device₁ 400) and a contact to another of the devices (device₂ 410). The modifications to the interactions and the channels can occur during a session 420 with an external customer 75.

Fig. 2 is a flow diagram depicting the steps in a method in which a service provider initiates an interaction with a customer. A customer profile database is provided in step 250. The customer profile database includes a plurality of dialogs 240 stored therein. A service provider 230 initiates an interaction with a customer in step 270. The specific customer in which an interaction is desired to be connected to is acquired by extracting/locating the customer from the customer profile database in step

260. The customer is identified to the interaction by updating the interaction in step 280. In step 290, the service provider initiates a connection to the customer and the interaction between the two entities begins. As the interaction proceeds, a service provider will either be prompted to take actions by the Content Analysis Object 130 or will take actions on its own. These actions are validated using criteria such as customer profile, privacy policy, business policy, channel limitations, and the subject matter of the interaction occurring. Once proposed actions by a service provider are authorized, the customer profile database may be updated in step 310.

Fig. 3 illustrates a flow diagram depicting a preferred embodiment wherein the connection of a service provider to a customer is determined based upon the subject matter of the interaction and the actions transpiring within the interaction. Identification of contact₁ and contact₂ are established in steps 1180 and 1190, respectively. Next, a first channel₁ and a second channel₂ are identified in steps 1200 and 1210, respectively. A first interaction₁ and a second interaction₂ are created in steps 1220 and 1230, respectively. A dialogue is established in step 1240. Customers are located in step 1270 and the interactions are evaluated to identify subjects associated with each of the interactions in steps 1250 and 1260, respectively. A first service provider₁ and a second service₂ are identified and associated with the interactions in steps 160 and 170, respectively. The service providers are selected based upon the subject matter of the interactions and the actions that are transpiring over the interactions such as technical questions from customer requiring the attention of a business's technical staff. Steps 180 and 190 validate all actions taken by the service providers to comport with business policies, customer profiles, channel

limitations, and customer privacy. Based on actions occurring within a dialogue the service provider may be prompted to take actions in steps 200 and 210. Finally, all actions occurring within the dialogue for each of the concurrent interactions are recorded as an update to the customer profile in
5 step 220.

Fig. 4 is a preferred architecture depicting the software objects and the object interfaces of the present invention. A Channel Director Object 10 is primarily responsible for coordinating and managing all Dialogue Objects 60 occurring within a business enterprise. The Channel Director
10 Object 10 binds multiple Interaction Objects 30 with a single Customer Object 70 into a single Dialogue Object 60. Customer Objects 70 in turn contain Privacy Objects 90 and Profile Objects 80. Privacy Objects include representations reflecting a particular Customer Object's 70 preferences such as preferred response channel; preferred payment method; name,
15 address, etc. Further, Privacy Objects 90 include representations of a Customer Object's 70 privacy requirements such as secured socket layer communication, password protection access, no email solicitations, etc.

Interaction Objects 30 are associated with a specific Customer Object 70 and a specific Channel Object 40. Interaction Objects 30 are established
20 by the Contact Manager Object 20. Electronic information occurring between a Customer Object 70 and a Service Provider Object 100 is recorded within a Trap Action Object 50 which is bound to an Interaction Object 30. These electronic information units are then communicated by the Dialogue Object 60 to a Content Analysis Object 130 which is bound to
25 a Dialogue Object 60. The Content Analysis Object 130 will evaluate the electronic information units recorded in the Trap Action Objects 50 for all Interaction Objects 30 occurring within a single Dialogue Object 60 using

the Privacy Object 90, the Profile Object 90, the Skills Object 110, and the Business Policy Object 120. Once evaluated, the Content Analysis Object 130 will validate any actions a Service Provider Object 100 attempts to take with a Customer Object 70 to ensure Business Policy Objects 120, customer
5 Profile Objects 80, customer Privacy Objects 90, and service provider Skills Objects 110 comport with any proposed action by the Service Provider Object 100. Additionally, the Content Analysis Object 130 will alert the Service Provider Object 100 to take certain actions based upon the evaluated subject matter of a Dialogue Object 60. Further, the Content
10 Analysis Object 130 may request that a different Service Provider Object 100 be bound to an Interaction Object 30 based upon the subject matter of an Interaction Object 30 or the actions occurring within an Interaction Object 30.

A Connection Manager Object 140 will establish initial connections
15 to Interaction Objects 30 and Service Provider Objects 100. It is also responsible for using the Business Policy Object 120 and the Skills Object 110 in selecting the most appropriate Service Provider Object 100 for a particular Interaction Object 30. The Connection Manager Object 140 also manages the load of the Service Provider Objects 100 to ensure overall
20 system performance.

A Business Policy Object 120 provides information related to things such as, but not limited to, customer wait time; maximum interactions per channel; customer dispute procedure; and customer refund policy. Each Service Provider Object 100 has a set of Skills Objects 110 which identify
25 the capabilities of the Service Provider Object 100 including such things as authorized channels, approval authority, technical capabilities, and maximum number of interactions permitted.

As one skilled in the art will appreciate, and as will be more fully explained below, the architecture described above will readily permit multiple occurring interactions with a customer over cross-channels to be normalized into a single electronic dialogue. The dialogue may then be
5 evaluated to perform a number of actions such as modifying the interaction; modifying the channel; providing near-term feedback to a customer by performing modifications during a session with the customer; and selecting a service provider based upon the interaction and the subject matter of the interaction.

10 This will permit business enterprises to enhance customer intimacy by substantially automating customer interactions so as to reduce and focus human intervention thereby reducing operating expenses. It further, provides for a complete 360 degree view of a customer with no interactions of a customer falling through the cracks when interactions occur over cross
15 channels within the business enterprise.

The remaining detailed description includes a description of each object; the object interfaces; and the methods available with each object.

The hierarchical view of the Channel Director Object 10 is illustrated in Fig. 5 along with the corresponding methods associated with the Channel
20 Director Object 10. The Channel Director Object 10 is the highest level object of a preferred embodiment depicted in Fig. 4. This object coordinates activities amongst the Contact Manager Object 20, a plurality of Dialogue Objects 60 and a connection Manager Object 140. Further, the Channel Director Object 10 permits the binding of initially anonymous
25 Interaction Objects 30 to an ongoing Dialogue Object 60 associated with a particular Customer Object 80. As Fig. 5 shows, the Channel Director

Object 10 instantiates a single Contact Manager Object 20, a plurality of Dialogue Objects 60, and a single Connection Manager Object 140.

The methods of the Channel Director Object 10 include a Contact Manager Object method 540, a Connection Manager Object method 570, an AddDialogue method 550, a RemoveDialogue method 560, a Shutdown method 580, and a Bind method 590. The Contact Manager Object method 540 and the Connect Manager Object method 570 simply permit a Contact Manager Object 20 and a Connection Manager Object 140 to be bound to the Channel Director Object 10. These derived objects will also be capable of accessing the Channel Director Object's 10 methods through normal inheritance. The AddDialogue method 550 and the RemoveDialogue method 560 permit Dialogue Objects 60 to be added and removed, respectively. The Shutdown method 580 permits a Channel Director 10 to gracefully close down which would permit recovery in the event of failures.

Finally, the Bind method 590 allows an Interaction Object 30 to bind with a particular Dialogue Object 60. This permits interactions occurring with an anonymous Customer Object 70 to continue as anonymous until the Interaction Action Object 30 can identify through the trapping of electronic messages and its corresponding Customer Object method 1160 (Fig. 14) the identity of the customer. Once the identity of the customer is known the Interaction Object 30 can join a Dialogue Object 60.

The Contact Manager Object 20 (Fig. 6) is derived from the Channel Director Object 10 and comprises Channel Objects 40 and Interaction Objects 30. The Contact Manager Object 20 is initiated such that it is aware of all the devices through which an external customer 75 may contact the business. Each device is associated with a channel that is initially bound to the Contact Manger Object 20. The Contact Manager Object 20

monitors the channels coming into an organization for a contact to be made. The Contact Manager Object 20 comprises an AddChannel method 600, an AddInteraction method 610, a SwitchChannel method 620, a RemoveChannel method 630, a RemoveInteraction method 640, a

5 LocateInteraction method 650, a LocateChannel method 660, and a ChannelLoad method 670. The AddInteraction 610 and AddChannel 600 methods permit the adding of Interaction Objects 30 and Channel Objects 40, respectively. The RemoveInteraction 640 and RemoveChannel 630 methods permit the removal of Interaction Objects 30 and Channel Objects

10 40, respectively. The SwitchChannel method 620 permits the Contact Manager Object 20 to change a channel associated with an Interaction Object 30. This permits an existing interaction which is taking place with a customer to cross channels within a single interaction. The

15 LocateInteraction 650 and LocateChannel 660 permit locating a particular Interaction Object 30 or Channel Object 40, respectively. Finally, the ChannelLoad method 670 allows the Contact Manager Object 20 to query a specific Channel Object 40 and determine the current load of that object, this will permit the Contact Manager Object 40 to maintain load balancing across all channels and permit a customer to switch channels should a

20 particular channel become idle or drop.

Fig. 7 illustrates a preferred Interaction Object 30. The Interaction Object 30 is derived from a Contact Manager Object 20. Accordingly the Interaction Object 30 has the operations associated with the Contact Manager Object 20 at its disposal through the properties of inheritance as

25 well as the operations of the Channel Director Object 10. The Interaction Object Comprises 30 a TrapAction Object 50, a Service Provider Object 100, a Customer Object 70 and a Channel Object 40. Fig. 7 further

illustrates a set of methods which are available with an Interaction Object 30. These methods comprise an ID method 1110, a Service Provider Object method 1120, a Channel Object method 1130, a Customer Object method 1140, an AddTrapAction method 1150, a CurrentPrivacy method 1160, a Status method 1170, a Subject method 1180, a CurrentPolicy method 1190, a CurrentProfile method 1200, and a GetTrapAction method 1210.

The ID method 430 returns a unique identifier associated with an Interaction Object 30. This method permits other objects to locate a particular interaction. Upon instantiation of an Interaction Object 30 an ID is created. The Service Provider Object method 440, the Channel Object method 450, and the Customer Object method 460 are nothing more than references to a Service Provider Object 100, a Channel Object 40, and a Customer Object 70, respectively. This permits the binding of these objects to a particular Interaction Object 30 and allows other objects to set or access these objects which are bound to a particular Interaction Object 30. The AddTrapAction method 470 permits electronic messages occurring between a customer and a service provider to be recorded with an Interaction Object 30. Correspondingly, the GetTrapAction method 530 permits other objects to access the last electronic message which occurred within an Interaction Object 30. This electronic message can then be stored or analyzed as discussed below by other objects. Further, a Status method 490 permits other objects in the system to query an Interaction Object 30 to determine the state that the Interaction Object 30. The states of an Interaction Object 30 comprise inactive, active, idle, and dropped. Also, a Subject method 500 permits the subject of an Interaction Object 30 to be set or queried. Finally, the CurrentPrivacy method 480, the CurrentPolicy 510, the CurrentProfile 520 correspond to customer privacy rules, business

policy rules, and customer profile rules, respectively. These rules are particular to an Interaction Object 30 and are retrievable by other objects when actions are being performed to ensure the appropriateness and validity of actions occurring within an Interaction Object 30.

5 The Dialogue Object 60 comprises Channel Objects 40, a Customer Object 70, Service Provider Objects 100, a Content Analysis Object 130, and Interaction Objects 30. Multiple Interaction Objects 30 will be associated with a single Dialogue Object 60. These interactions are bound to the Dialogue Object 60 through the Channel Director Object's 10 Bind
10 590 method. Initially, an Interaction Object 30 may not be associated with a particular Customer Object 70, but once enough information contained in the TrapAction Object 50 exists to permit Customer Object 70 identification, the Interaction Object 30 will call the Channel Director Object's 10 Bind method 590 to bind to the Dialogue Object 60. When this
15 binding occurs the appropriate Dialogue Object 60 will issue an AddInteraction 730 call which will then permit the Dialogue Object 60 to manage the Interaction Object 30 and keep track of electronic messages which are occurring within the Interaction Object 30 via the TrapAction Object 50.

20 Referring again to the Dialogue Object 60, an action (or electronic communication) occurring inside the Interaction Object 30 via the TrapAction Object 50 will cause an event to be raised to the Dialogue Object 60 which will precipitate an analysis to occur inside the Content Analysis Object 130. The Dialogue Object 60 is aware of all Interaction
25 Objects 40 occurring within an dialogue and will issue a call to the Content Analysis Object 130 when the Dialogue Object's 60 has its ActionOccurred 810 method set.

A preferred Content Analysis Object 130 is shown in Fig. 9. The Content Analysis Object 130 is responsible for evaluating actions which are transpiring within a Dialogue Object 60. The Dialogue Object 60 knows from which interaction a particular action is occurring. The Content

5 Analysis Object 130 records the actions and evaluates the actions to determine if subjects can be identified within the interaction and ensures the validity of actions that take place within an interaction. Business policies, customer profiles, and customer privacy will be enforced against any action which a Service Provider Object 100 attempts to take. In

10 addition, the Content Analysis Object 130 can be used to change an existing Service Provider Object 100 by issuing an appropriate command. This object may also prompt, based upon business policy, a Service Provider Object 100 into taking a particular action with a Customer Object 70 during an interaction.

15 The Content Analysis Object 130 primarily interfaces with the Dialogue Object 60, the Connection Manager Object 140, and the Business Policy Object 120 as shown in Fig. 4. The object hierarchy of the Content Analysis Object 130 is shown in Fig. 9 and comprises an Interaction Object 30, a Connection Manager Object 140, a Business Policy Object 120, and

20 Trap Action Objects 50. Fig. 9 also depicts the methods available with the Content Analysis Object 130. These methods comprise a Connection Manager Object method 1030, an Interaction Object method 1040, a Business Policy Object method 1050, an AddTrapAction method 1060, an Evaluate method 1070, and a TakeAction method 1080. The Connection

25 Manager Object method 1030, the Interaction Object method 1040, and the Business Policy Object method 1050 permit the binding of the Connection Manager Object 140, the Interaction Object 30, and the Business Policy

Object 120, respectively. Recording of electronic messages occurs within Trap Action Objects 50 and correspondingly the AddTrapAction method 1060 permits these messages to be stored for an entire Dialogue Object 60 which has multiple Interaction Objects 30 contained therein. The Evaluate
5 method 1070 needs to perform a variety of functions.

Preferably, the Evaluate method 1070 will issue a new query upon each added Trap Action Object 50 to acquire a customer's Profile Object 80, Privacy Object 90 and the Business Policy Object 120. The Interaction Object 30 has methods as defined above which will define the customer and
10 the current profile as well as the current privacy. The format of this information is discussed below with the Business Policy Object 120. This information combined with the actions occurring in the Trap Action Objects 50 will permit the Evaluate method 1070 to ensure actions which are initiated from a Service Provider Object 100 conform to business policy,
15 customer profiles, and privacy requirements. Further, actions taken by an external customer 75 will often identify specific pre-defined subjects. Subjects can be identified in a plurality of situations such as being explicitly identified by a Service Provider Object 100, and set by a Customer Object 70 by traversing a web link whose address (URL)
20 identifies a subject, such as "buyproducts.html." Further, as one skilled in the art would readily appreciate a variety of intelligent software agents could be used to identify the subject matter of electronic correspondences by using computational linguistic techniques to evaluate the underlying electronic correspondence. This identification of an Interaction Object's 30
25 subject matter will permit the Evaluate method 1070 to switch to a new Service Provider Object 100 if necessary or to prompt the existing Service Provider Object 100 to take a specific action such as call the help desk for

support. The Skills Object 110 will identify subjects which a Service Provider Object 100 is conversant.

Finally, in Fig. 9 the Content Analysis Object 130 has a TakeAction method 1080 which permits communication to the Dialogue Object 60.

- 5 This communication authorizes the Dialogue Object 60 to record and relay the action which the Content Analysis Object 130 is issuing to a Service Provider Object 100 which is bound to a particular Interaction Object 30 occurring within a single Dialogue Object 60.

- Fig. 10 is a preferred Customer Object 70 in accordance with the present invention. The Customer Object 70 is responsible for maintaining customer information such as customer profiles and customer privacy requirements. If a specific customer is not identified, when a Customer Object 70 is initiated, it is an anonymous Customer Object 70. Upon queries issued to the anonymous Customer Object 70, it will transform
10 itself into a particular Customer Object 70 and have available the customer profile and customer privacy at its disposal. In this way, an anonymous Customer Object 70 can also be used to add a new customer to the existing customers of the business through use of methods which permit a user of the Customer Object 70 to publish information. In the preferred
15 embodiment, the Customer Object 70 has access to and knowledge of the customer profile database for the business organization thereby permitting the above disclosed functionality.

- Referring to Fig. 10 a Customer Object 70 has an object hierarchy comprised of a Profile Object 80 and a Privacy Object 90. These objects
25 are simple objects which interface with the underlying customer profile database and return fielded strings of numeric text as discussed below with the Business Policy Object 120. The customer profile is available to users

of the Customer Object 70 via the GetProfile 900 method. The privacy information about a customer is available through the GetPrivacy 910 method. Correspondingly, a customer profile and customer privacy may be published and stored through the PublishProfile method 930 and the PublishPrivacy method 940, respectively. Further, a PublishCustomer method 920 permits a new customer to be created for the business enterprise. Finally, a Locate method 890 permits an anonymous Customer Object 70 to take a search query and use that query to find a particular customer in the customer profile database.

The Trap Action Object 50 is another simple object which simply converts the keystrokes and mouse clicks of computer screens into a fielded numeric string as discussed below with the Business Policy Object 120. This normalization of the electronic messages transpiring between a customer and a service provider permits the Content Analysis Object 130, as described above, to evaluate and take actions so as to modify the interactions, the channels, and assist in selecting a service provider on the basis of the subject matter of the interaction.

As one skilled in the art will readily appreciate the Content Analysis Object 130 and its corresponding Evaluate method 1070 will permit a variety of customized responses to a customer including modifying an interaction by forcing an action on the Service Provider Object 100; selecting a Service Provider Object 100 on the basis of an interaction's subject matter; enforcing privacy, customer profiles, and business policy on interactions taken from a Service Provider Object 100; alerting the Service Provider Object 100 to take a particular action; and modifying a channel should an interaction request such an action.

The Channel Object 40 associated with an added Interaction Object 30 can be acquired from the Interaction Object's 30 Channel Object method 450. The Channel Object 40 and its methods are illustrated in Fig. 11.

Channel Objects 40 are responsible for maintaining information about its own type such as status, load, and number of interactions. In this way a

Channel Object 40 can manage the channel within which a customer is interacting and ensure that proper performance is being achieved. The

Channel Object 40 comprises a Type method 680, a Status method 690, a NotifySwitch method 700, and a Load method 710. The Type method 680

indicates what device is associated with a particular Channel Object 40, devices may include such things as fax, Internet, cellular, kiosk, walk-ups, etc. Each Channel Object 40 may be queried to determine its status and

load through the Status 690 and Load 710 methods. The status of a Channel Object 40 comprise the states of inactive, active, idle, and dropped. The

load of a Channel Object 40 is maintained by channel type and indicates the current usage percentage of the available channels of a particular type. For

example, a Channel Object 40 of type Internet would provide a Load method 710 which when called would return a percentage. The percentage would indicate the usage of the Internet channel type. As one skilled in the

art would readily appreciate, this would provide for easy load balancing between channels existing within the Channel Director Object 10. Finally,

the NotifySwitch method 700 permits the Channel Object 40 to raise a signal to the Contact Manager Object 20 that a particular channel needs to be changed. In this way the Contact Manager Object 20 may switch

Interaction Objects 30 accordingly with minimal interruption to the external customer 75 who is communicating over a particular channel.

The Service Provider Object 100 is illustrated in Fig. 12. As previously indicated, a Service Provider Object is an electronic representation of a business service provider. It is through this object that a business will interact with a customer via the Interaction Object 30. A
5 Service Provider Object 100 may be an electronic representation of a human entity or an electronic service provider such as an automated email service provider. A single Service Provider Object 100 may be interacting with multiple Customer Objects 70 at one time.

The Service Provider Object 100 comprises Skills Objects 100,
10 Interaction Objects 30, and Channel Objects 40. The Service Provider Object 100 further comprises an AddInteraction method 950, a RemoveInteraction Object 960, an AddChannel method 970, a RemoveChannel method 980, a GetSkills method 990, a PublishSkills method 1000, a TotalInteractions method 1010, and a TakeAction method
15 1020.

The Skills Object 110 is another basic object which has access and interfaces to the business enterprise's service provider database. It will include attributes associated with each business Service Provider Object 100 such as authorized channels, conversant subjects, technical abilities,
20 etc.

Fig 13, illustrates a Business Policy Object 120. The Business Policy Object 120 consists of Customer Objects 70 and Service Provider Objects 100. The Business Policy Object 120 stores and retrieves business policy, customer profiles, and may select a particular Service Provider Object 100
25 which should be used given a customer profile, and skill set of a Service Provider Object 100. The Business Policy Object comprises a GetPolicy method 850, a GetProfile method 860, a SelectServiceProvider method 870,

and a PublishPolicy method 880. The GetPolicy method 850 and the GetProfile method 860 return a business policy and a customer profile, respectively. The particular format of business policies, customer profiles, and the skill sets of service providers would be readily apparent to those skilled in the art and will vary based upon the needs of a particular business. In the preferred embodiment, this information is all contained in a fielded string, the contents of the fielded string being numeric numerals. These numerals are then looked up in a flat table to identify readable text which may be displayed when appropriate to the user. For example, a business policy of no more than 25 interactions per Internet channel and a requirement that each Internet Service Provider Object 100 respond to a request within 2 minutes will be encoded in a business policy fielded string as "0120," where an external table indicates that "01" is the policy for the interaction limits on the Internet channel, and "20" indicates that Internet Service Provider Object 100 must respond within 2 minutes of a customer request. Similarly, the skill sets of a service provider, customer profile, customer privacy, Trap Action Objects 50, and actions are preferably represented as described above. The SelectServiceProvider method 870 will use the skill sets of the Service Provider Object 100, the business policy, and the customer profile to select an appropriate Service Provider Object 100 when requested to do so. This method returns a Service Provider Object 100. Finally, the PublishPolicy method 880 permits policies to be updated and stored as needed.

A final Connection Manager Object 140 is provided in Fig. 14. The Connection Manager Object 140 is responsible for initially binding an Interaction Object 30 to a specific Service Provider Object 100. It will use the Business Policy Object 120 in its selection process. Furthermore, the

Connection Manager Object 140 will monitor all Service Provider Objects 100 within the Channel Director 10 to ensure proper load balancing and performance of Service Provider Objects 100. The Connection Manager Object 140 binds to Interaction Objects 30, the Business Policy Object 120,
 5 and the Service Provider Objects 100.

The methods of the Connection Manager Object 140 include a Status method 1040, an AddServiceProvider method 1100, a RemoveServiceProvider method 1110, a Switch method 1120, a LoadServiceProvider method 1130, and AddInteraction method 1140, and a
 10 RemoveInteraction method 1150. The AddServiceProvider 1100, RemoveServiceProvider 1110, AddInteraction 1140, and RemoveInteraction 1150 methods permit the Connection Manager Object 140 to manage Service Provider Objects 100 and Interaction Objects 30 by adding and deleting them from internally kept queues. The LoadServiceProvider
 15 method 1130 issues a query to a specific Service Provider Object 100 to determine current capacity such as number of existing interactions with Customer Objects 70. The load of a Service Provider Object 100 may dictate that a switch needs to occur to a different less burdened Service Provider Object 100. This is provided for with a Switch method 1120.
 20 Finally, a Status method 1090 keeps track of the state of the Connection Manager Object 140 which may be needed on occasion by the Channel Director Object 10 or in the event of failures. This Status method 1090 contains values similar to those discussed above with the Interaction Object 30.

25 The foregoing description of preferred embodiments of the invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form

disclosed, and obviously many modifications and variations are possible in light of the above teachings. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application to thereby enable one skilled in the art to best utilize the

5 invention in various embodiments and with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the claims appended hereto.

007220-20504560